

Solving DC programs with polyhedral component utilizing a multiple objective linear programming solver

Andreas Löhne *

Andrea Wagner †

October 19, 2016

Abstract

A class of non-convex optimization problems with DC objective function and linear constraints is studied, where DC stands for being representable as the difference $f = g - h$ of two convex functions g and h . In particular, we deal with the special case where one of the two convex functions g or h is polyhedral. In case of g being polyhedral, we show that a solution of the DC program can be obtained from a solution of an associated polyhedral projection problem. In case of h being polyhedral, we prove that a solution of the DC program can be obtained by solving a polyhedral projection problem and finitely many convex programs. Since polyhedral projection is equivalent to multiple objective linear programming (MOLP), a MOLP solver (in the second case together with a convex programming solver) can be used to solve instances of DC programs with polyhedral component. Numerical examples are provided, among them an application to locational analysis.

Keywords: DC programming, global optimization, polyhedral projection, multiple objective linear programming, linear vector optimization

MSC 2010 Classification: 15A39, 52B55, 90C29, 90C05

1 Introduction

A function $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ is said to be *polyhedral convex* if its epigraph $\text{epi } f := \{(x, r) \in \mathbb{R}^n \times \mathbb{R} \mid r \geq f(x)\}$ is a polyhedral convex set. We consider the following DC optimization problem

$$\min [g(x) - h(x)] \text{ subject to } x \in \text{dom } g, \quad (\text{P})$$

where we assume that $g : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ and $h : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ are convex and at least one of the functions g and h is polyhedral convex. DC optimization problems are investigated for instance in [1, 10, 16, 20, 22, 25, 31, 32].

The methods presented in this article are based on computing the vertices of $\text{epi } g$ in case of g is polyhedral and the vertices of $\text{epi } h^*$ in case of h is polyhedral, where h^* is the conjugate of h . It is shown that a solver for multiple objective linear programs (MOLP) can be used for this task. The aim of this paper is to show that global solutions of the considered class

*Friedrich Schiller University Jena, Department of Mathematics, 07737 Jena, Germany, andreas.loehne@uni-jena.de

†Vienna University of Economics and Business, Institute for Statistics and Mathematics, andrea.wagner@wu.ac.at

of DC-programs can be obtained by utilizing a MOLP-solver. The main advantage of this procedure is that the algorithms are very easy to implement. Nevertheless, the algorithms can be applied to certain non-convex problems in location theory, discussed below. We will show that they are competitive to some algorithms from the recent literature. It should be noted that the presented methods can be combined with other global optimization techniques in order to avoid to compute all vertices of the epigraphs. But then, a modification of the MOLP-solver is required, which would make the implementation more difficult.

The computation of vertices of an epigraph is related to the problem to project a polyhedron into a subspace. A *polyhedral convex set* P in \mathbb{R}^n (or *polyhedron* for short) is defined as the solution set of a system of finitely many linear inequalities, that is,

$$P = \{x \in \mathbb{R}^n \mid Bx \geq c\} \quad (1)$$

for a matrix $B \in \mathbb{R}^{m \times n}$ and a vector $c \in \mathbb{R}^m$. The pair (B, c) is said to be an *H-representation* of P . The well-known Weyl-Minkowski theorem states that every polyhedron can be expressed as the *generalized convex hull* of finitely many points $v^i \in \mathbb{R}^n$; $i = 1, \dots, r$; $r \geq 1$ and finitely many directions $d^j \in \mathbb{R}^q \setminus \{0\}$; $j = 1, \dots, s$; $s \geq 0$, that is

$$P = \{x \in \mathbb{R}^n \mid x = V\lambda + D\mu, e^T \lambda = 1, \lambda \geq 0, \mu \geq 0\}, \quad (2)$$

where v^i and d^j are the columns of $V \in \mathbb{R}^{n \times r}$ and $D \in \mathbb{R}^{n \times s}$, respectively. Further we denote by $e := (1, \dots, 1)^T$ the all-one-vector. We say that (V, D) is a *V-representation* of the polyhedron P . Both H-representation and V-representation are special cases of the *projection representation* or *P-representation* (B, C, c) , that is

$$P = \left\{x \in \mathbb{R}^n \mid \exists u \in \mathbb{R}^k : Bx + Cu \geq c\right\}, \quad (3)$$

for matrices $B \in \mathbb{R}^{m \times n}$, $C \in \mathbb{R}^{m \times k}$, $c \in \mathbb{R}^{m \times 1}$. In Sections 2 and 7 we will see that in many situations only a P-representation of a polyhedron is known. An H-representation can be obtained from a P-representation by Fourier-Motzkin elimination and a V-representation can be obtained from an H-representation by vertex enumeration, for instance, by using the double description method by Motzkin et al. [33]. However, the first part of this procedure (Fourier-Motzkin elimination) is not tractable if k is large. It has been shown in [28] that both an H-representation and a V-representation can be obtained from a P-representation by solving a multiple objective linear program (MOLP). This follows from the equivalence between a polyhedral projection problem (which is roughly speaking the problem to obtain a V-representation from a P-representation (3)) and multiple objective linear programming, as shown in [28]. By *equivalence* the authors of [28] understand that a solution of the one problem can be “easily” (for details see [28] or Section 6 below) obtained from a solution of the other problem. In order to compute a V-representation of (3), a multiple objective linear program with $n + 1$ objective functions needs to be solved.

Multiple objective linear programs can be solved by Benson’s algorithms [2]. Numerical examples for up to 10 objective functions are provided in [5] and [29]. A more direct way to solve the polyhedral projection problem is the *convex hull method* by Lassez and Lassez [24]. This method, however, is closely related to the dual variant of Benson’s algorithm for multiple objective linear programs [12], which has been developed independently.

We close this section with some notation. We denote by v_i the components of a vector v and by M^i the rows of a matrix M . As mentioned above, we set $e = (1, \dots, 1)^T$. Given a

convex function $g : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ we denote by $\text{dom } g := \{x \in \mathbb{R}^n \mid g(x) \neq \infty\}$ the domain of g , and by $g^* : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$, $g^*(x^*) := \sup_{x \in \text{dom } g} [x^{*T}x - g(x)]$ the conjugate of g . The recession cone of a polyhedron (1) is the polyhedral convex cone $0^+P := \{x \in \mathbb{R}^n \mid Bx \geq 0\}$. The closure, interior, boundary of a set A is denoted, respectively, by $\text{cl } A$, $\text{int } A$, $\text{bd } A$.

2 Representing polyhedral convex functions

Polyhedral convex functions can be represented by their epigraphs. The following definition is based on a P-representation of the epigraph.

Definition 1. A matrix $A \in \mathbb{R}^{m \times (n+1+k+1)}$ is called a representation of a polyhedral convex function $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ if

$$\text{epi } f = \left\{ \begin{pmatrix} x \\ r \end{pmatrix} \in \mathbb{R}^{n+1} \mid \exists u \in \mathbb{R}^k : A \begin{pmatrix} x \\ r \\ u \\ -1 \end{pmatrix} \geq 0 \right\}. \quad (4)$$

If the matrix A is partitioned as $A = (B, b, C, c)$, where $B \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $C \in \mathbb{R}^{m \times k}$ and $c \in \mathbb{R}^m$, we obtain

$$\text{epi } f = \left\{ \begin{pmatrix} x \\ r \end{pmatrix} \in \mathbb{R}^{n+1} \mid \exists u \in \mathbb{R}^k : Bx + br + Cu \geq c \right\}. \quad (5)$$

The following examples motivate the definition of a representation. We show that for many (classes of) polyhedral convex functions a representation can be obtained without any essential computational effort.

Example 2. Consider finitely many affine functions $f_i(x) = D^i x + d_i$ ($i = 1 \dots, m$), a matrix $P \in \mathbb{R}^{k \times n}$ and a vector $p \in \mathbb{R}^k$. Then

$$f(x) := \begin{cases} \max_{i \in \{1, \dots, m\}} f_i(x) & \text{if } Px \geq p \\ +\infty & \text{otherwise} \end{cases}$$

is a polyhedral convex function with representation

$$A = \begin{pmatrix} -D & e & 0 & d \\ P & 0 & 0 & p \end{pmatrix},$$

where D^1, \dots, D^m are the rows of $D \in \mathbb{R}^{m \times n}$ and $d := (d_1, \dots, d_m) \in \mathbb{R}^m$.

Example 3. Let f be a polyhedral convex function and let (V, D) be a V-representation of $\text{epi } f$, that is, $V \in \mathbb{R}^{(n+1) \times r}$ ($r \geq 1$), $D \in \mathbb{R}^{(n+1) \times s}$ ($s \geq 0$),

$$\text{epi } f := \{z \in \mathbb{R}^{n+1} \mid \exists \lambda \in \mathbb{R}^r, \mu \in \mathbb{R}^s : z = V\lambda + D\mu, e^T \lambda = 1, \lambda \geq 0, \mu \geq 0\}.$$

Then

$$A = ((B, b) \quad C \quad c) = \begin{pmatrix} \mp I & (\pm V, \pm D) & 0 \\ 0 & I & 0 \\ 0 & (\pm e^T, 0) & \pm 1 \end{pmatrix}$$

is a representation of f .

Example 4. Let $g, h : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ be polyhedral convex functions with representations

$$A_g = \begin{pmatrix} B_g & b_g & C_g & c_g \end{pmatrix} \quad \text{and} \quad A_h = \begin{pmatrix} B_h & b_h & C_h & c_h \end{pmatrix}.$$

The *infimal convolution* of g and h is the polyhedral convex function f with

$$\text{epi } f = \text{epi } g + \text{epi } h.$$

Thus, its representation is $A_f = \begin{pmatrix} B_f & b_f & C_f & c_f \end{pmatrix}$ with

$$\begin{pmatrix} B_f & b_f \end{pmatrix} = \begin{pmatrix} \pm I \\ 0 \\ 0 \end{pmatrix} \quad C_f = \begin{pmatrix} \mp I & 0 & \mp I & 0 \\ (B_g \ b_g) & C_g & 0 & 0 \\ 0 & 0 & (B_h \ b_h) & C_h \end{pmatrix} \quad c_f = \begin{pmatrix} 0 \\ c_g \\ c_h \end{pmatrix}.$$

Example 5. Let a representation $A = \begin{pmatrix} B & b & C & c \end{pmatrix}$ of a polyhedral convex function $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ be given, then

$$A^* = \begin{pmatrix} \pm I & 0 & \pm B^T & 0 \\ 0 & 0 & \pm C^T & 0 \\ 0 & 0 & \pm b^T & \pm 1 \\ 0 & 1 & c^T & 0 \\ 0 & 0 & I & 0 \end{pmatrix}$$

is a representation of the conjugate $f^* : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ of f . The details are shown in the following proposition.

Proposition 6. *For a polyhedral convex function $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$, the following two statements are equivalent:*

$$\text{epi } f = \left\{ \begin{pmatrix} x \\ r \end{pmatrix} \in \mathbb{R}^{n+1} \mid \exists u \in \mathbb{R}^k : Bx + br + Cu \geq c \right\}; \quad (6)$$

$$\text{epi } f^* = \left\{ \begin{pmatrix} x^* \\ r^* \end{pmatrix} \in \mathbb{R}^{n+1} \mid \exists v \in \mathbb{R}_+^m : B^T v + x^* = 0, \ b^T v = 1, \ C^T v = 0, \ c^T v + r^* \geq 0 \right\}. \quad (7)$$

Proof. Assume that (6) is satisfied. Let $(x^*, r^*) \in \text{epi } f^*$, i.e.,

$$r^* \geq f^*(x^*) = \sup_{x \in \mathbb{R}^n} (x^{*T} x - f(x)) = \sup_{(x,r) \in \text{epi } f} (x^{*T} x - r).$$

This is equivalent to the implication

$$(x, r) \in \text{epi } f \implies r^* \geq x^{*T} x - r.$$

By (6), this can be expressed as

$$Bx + br + Cu \geq c \implies r^* \geq x^{*T} x - r.$$

By Farkas' lemma (see e.g. [18, Theorem 7.20]) we obtain the existence of $v \in \mathbb{R}_+^m$ such that

$$B^T v + x^* = 0, \ b^T v = 1, \ C^T v = 0, \ c^T v + r^* \geq 0,$$

i.e., (7) holds. The opposite implication follows likewise by taking into account that $f = f^{**}$ for a polyhedral convex function f . \square

The list of examples could be extended. Note that, in general, it is much more expensive to compute an H-representation or a V-representation of the epigraph of a polyhedral convex function. This problem, however, is important for our studies and will be discussed in Section 6.

3 Maximizing a convex function over a polyhedron

A well-known result on maximizing a convex function over a polytope P is the attainment of the maximum at least in some vertex of P . We discuss in this short section the case of unbounded polyhedra.

Let $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ be convex and let P be a polyhedron. Assume that P has a vertex. We consider the problem

$$\max f(x) \quad \text{subject to} \quad x \in P. \quad (\text{C})$$

Proposition 7. *If Problem (C) has an optimal solution, then some vertex x^* of P is an optimal solution of (C).*

Proof. We denote by v^1, \dots, v^k the vertices of P and set $B := \text{conv} \{v^1, \dots, v^k\}$. Convexity of f implies that there is a vertex v^j of B which is an optimal solution of

$$\max f(x) \quad \text{subject to} \quad x \in B.$$

Assume there is $x \in P \setminus B$ with $f(x) > f(v^j)$. The point x can be represented as the sum $x = b + d$ of a point $b \in B$ and a direction $d \in (0^+P) \setminus \{0\}$. Then, $f(b + d) > f(v^j) \geq f(b)$. Convexity of f implies that $f(b + nd) \rightarrow \infty$ as $n \rightarrow \infty$, which contradicts the existence of an optimal solution. \square

Then next statement characterizes the existence of a solution.

Proposition 8. *Problem (C) is unbounded if and only if there exists a vertex v of P , an extreme direction d of P and some $\beta > 0$ with*

$$f(v) < f(v + \beta d).$$

Proof. Let (C) be unbounded. There is a sequence (x_ν) in P with $(f(x_\nu)) \rightarrow \infty$ as $\nu \rightarrow \infty$. For all $\nu \in \mathbb{N}$, we have

$$x_\nu = \sum_{i=1}^r \lambda_i^\nu v^i + \sum_{j=1}^s \mu_j^\nu d^j, \quad \sum_{i=1}^r \lambda_i^\nu = 1, \quad \lambda_i^\nu \geq 0, \quad \mu_j^\nu \geq 0,$$

where v^i are the vertices and d^i the extreme directions of P . Setting

$$\alpha_{ij}^\nu := \frac{\lambda_i^\nu \mu_j^\nu}{\beta^\nu} \quad \text{and} \quad \beta^\nu := \sum_{k=1}^s \mu_k^\nu,$$

we obtain

$$x_\nu = \sum_{i=1}^r \sum_{j=1}^s \alpha_{ij}^\nu (v^i + \beta^\nu d^j), \quad \sum_{i=1}^r \sum_{j=1}^s \alpha_{ij}^\nu = 1, \quad \alpha_{ij}^\nu \geq 0.$$

By Proposition 7, we have $\beta^\nu \neq 0$ for ν being sufficiently large. Convexity implies that

$$\sum_{i=1}^r \sum_{j=1}^s \alpha_{ij}^\nu f(v^i + \beta^\nu d^j)$$

tends to infinity. Hence $f(v^i + \beta^\nu d^j) \rightarrow \infty$ for at least one i and j , which proves the first part of the statement.

The converse implication follows from convexity of f along the ray $\{v + \beta d \mid \beta \geq 0\}$, which belongs to P . \square

4 The case of g being polyhedral

The algorithm considered in this section is based on an enumeration of the vertices of $\text{epi } g$. We start with a reformulation of the DC program (P).

Proposition 9. *Problem (P) can be expressed equivalently as*

$$\min[r - h(x)] \quad \text{subject to} \quad (x, r) \in \text{bd epi } g. \quad (8)$$

Proof. Since $g : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ (i.e. the value $-\infty$ is not allowed), for the graph $\text{gr } g := \{(x, g(x)) \mid x \in \text{dom } g\}$ of g , we have

$$\text{gr } g \subseteq \text{bd epi } g \subseteq \text{epi } g = \text{gr } g + (\{0\} \times \mathbb{R}_+).$$

Let \bar{x} be an optimal solution of (P). For all $(x, r) \in \text{epi } g$ (and in particular for all $(x, r) \in \text{bd epi } g$) we have $g(\bar{x}) - h(\bar{x}) \leq g(x) - h(x) \leq r - h(x)$. Since $(\bar{x}, g(\bar{x})) \in \text{bd epi } g$, we conclude that $(\bar{x}, g(\bar{x}))$ solves (8).

Vice versa, let $(\bar{x}, \bar{r}) \in \text{bd epi } g$ be a solution of (8), then

$$\forall (x, r) \in \text{gr } g \subseteq \text{bd epi } g : \quad \bar{r} - h(\bar{x}) \leq r - h(x).$$

We get $\bar{r} = g(\bar{x})$ and see that $\bar{x} \in \text{dom } g$ solves (P). □

Assume now that $g : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ is polyhedral convex. Let $\{F_i \mid i = 1, \dots, k\}$ be the (finite) set of all facets of $\text{epi } g$. Then, it is evident that

$$\text{bd epi } g = \bigcup_{i=1, \dots, k} F_i.$$

The following statement is now obvious.

Corollary 10. *For $i = 1, \dots, k$, let (x^i, r^i) be an optimal solution of*

$$\min[r - h(x)] \quad \text{subject to} \quad (x, r) \in F_i. \quad (\text{P}_i)$$

Let $j \in \text{argmin} \{r^i - h(x^i) \mid i = 1, \dots, k\}$. Then, (x^j, r^j) is an optimal solution of (P). Problem (P) has an optimal solution if and only if, for all $i \in \{1, \dots, k\}$, (P_i) has an optimal solution.

Assume that (P) has an optimal solution. Then, for every $i \in \{1, \dots, k\}$, Problem (P_i) has an optimal solution which is, by Proposition 7, a vertex of the facet F_i of $\text{epi } g$. Since a vertex of a facet F of $\text{epi } g$ is also a vertex of $\text{epi } g$, an optimal solution of (P) can be always found among the vertices of $\text{epi } g$, denoted $\text{vert epi } g$.

Corollary 11. *Assume that (P) has an optimal solution. Every optimal solution (\bar{x}, \bar{r}) of*

$$\min[r - h(x)] \quad \text{subject to} \quad (x, r) \in \text{vert epi } g. \quad (9)$$

is also an optimal solution of (P).

If a representation A of the polyhedral convex function g is given, the vertices of $\text{epi } g$ can be computed by solving a polyhedral projection problem, see Section 6. If $\text{vert epi } g$ has been computed, it remains to determine the minimum over finitely many points in problem (9). We obtain the following solution procedure for the DC optimization problem (P).

Assumptions:

- (i) (P) has an optimal solution,
- (ii) g is polyhedral convex,
- (iii) $\text{epi } g$ has a vertex,
- (iv) a representation A of g is known.

Algorithm 12. (1) Compute the vertices $(x^1, r^1), \dots, (x^k, r^k)$ of $\text{epi } g$ by solving a polyhedral projection problem, see Section 6.

(2) Choose (x^j, r^j) such that $r^j - h(x^j) = \min_{i=1, \dots, k} [r^i - h(x^i)]$.

Result:

x^j is an optimal solution of (P).

Corollary 13. Algorithm 12 works correctly if the assumptions (i)-(iv) are satisfied.

Proof. This follows from the considerations above. □

5 The case of h being polyhedral

We assume now that the function h in Problem (P) is polyhedral. We consider the Toland-Singer dual problem of (P), see [36, 37], that is,

$$\min_{y \in \text{dom } h^*} [h^*(y) - g^*(y)], \quad (\text{D})$$

where $h^*(y) := \sup_{x \in \mathbb{R}^n} [y^T x - h(x)]$ is the conjugate of h and likewise for g . Due to the duality theory by Toland and Singer the optimal objective values of (P) and (D) coincide in case that the function h is closed. This condition is satisfied as we assume h to be a polyhedral convex function.

Since g^* is convex and h^* is polyhedral convex, Problem (D) can be solved by applying Algorithm 12 in Section 4. As pointed out below, a solution of (P) can be computed from a solution of (D). We suppose the following assumptions, all being related to the given problem (P):

Assumptions:

- (i') (P) has an optimal solution,
- (ii') h is polyhedral convex,

- (iii') $\dim \operatorname{epi} h = n + 1$,
- (iv') a representation A of h is known,
- (v') g is closed.
- (vi') A solution of the following problem exists whenever the problem is bounded for the parameter $y \in \mathbb{R}^n$:

$$\min_{x \in \mathbb{R}^n} [g(x) - y^T x]. \quad (10)$$

The following result relates solutions of (P) and solutions of (D) to each other. We denote by $\partial f(\bar{x}) := \{y \in \mathbb{R}^n \mid \forall x \in \mathbb{R}^n : f(x) \geq f(\bar{x}) + y^T(x - \bar{x})\}$ the *subdifferential* of a convex function $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ at $\bar{x} \in \operatorname{dom} f$. We set $\partial f(\bar{x}) := \emptyset$ for $\bar{x} \notin \operatorname{dom} f$. For convenience of the reader and in regards to Remark 15 below, we provide a short prove of the following statement.

Proposition 14. *(e.g. [23, Proposition 4.7] or [38, Proposition 3.20]) Let $g : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ and $h : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ be proper convex functions. Then the following holds true.*

(i) *If x is an optimal solution of (P), then each $y \in \partial h(x)$ is an optimal solution of (D).*

If in addition g and h are closed, a dual statement holds:

(ii) *If y is an optimal solution of (D), then each $x \in \partial g^*(y)$ is an optimal solution of (P).*

Proof. To prove statement (ii), we note that we have

$$\forall y^* \in \operatorname{dom} h^* : h^*(y^*) - g^*(y^*) \geq h^*(y) - g^*(y), \quad (11)$$

$$g(x) + g^*(y) = y^T x, \quad (12)$$

where the latter equation requires g being closed, compare [35, Theorem 23.5]. Moreover we use Fenchel's inequality, that is, $g^*(y^*) + g(x) \geq y^{*T}x$ for all $x \in \mathbb{R}^n$ and all $y^* \in \mathbb{R}^n$ (and likewise for h). For all $y^* \in \operatorname{dom} h^*$ and all $x \in \operatorname{dom} g$ we get

$$\begin{aligned} g(x) + h^*(y^*) - y^{*T}x &\geq h^*(y^*) - g^*(y^*) \stackrel{(11)}{\geq} h^*(y) - g^*(y) \\ &\geq y^T x - h(x) - g^*(y) \stackrel{(12)}{=} g(x) - h(x). \end{aligned}$$

Taking the infimum over $y^* \in \operatorname{dom} h^*$, we get

$$\forall x \in \operatorname{dom} g : g(x) - h^{**}(x) \geq g(x) - h(x).$$

The claim follows as $h^{**}(x) = h(x)$. Here we use that h is assumed to be closed, compare [35, Theorem 12.2]. The proof of (i) is analogous but neither g nor h needs to be closed, compare [35, Theorem 23.5] and take into account the fact that no biconjugation argument is necessary. \square

Remark 15. In [23, Proposition 4.7] the assumption of g and h being closed for statement (ii) of Proposition 14 is missing. Moreover in [38, Proposition 3.20] closedness of g needs to be assumed. Indeed, consider the following example. Let

$$f : \mathbb{R} \rightarrow \mathbb{R} \cup \{\infty\}, \quad f(x) := \begin{cases} \infty & \text{if } x < 0 \\ 1 & \text{if } x = 0 \\ 0 & \text{if } x > 0. \end{cases}$$

Then f is not closed and we have

$$f^* : \mathbb{R} \rightarrow \mathbb{R} \cup \{\infty\}, \quad f^*(y) := \begin{cases} 0 & \text{if } y \leq 0 \\ \infty & \text{if } y > 0. \end{cases}$$

Set $g = f$ and $h = \text{cl } f$. Then $y = 0$ solves (D) and $0 \in \partial g^*(0)$. But $x = 0$ is not a solution of (P). This shows the failure when g is not closed.

Set $g = \text{cl } f$ and $h = f$. Then $y = 0$ solves (D) and $1 \in \partial g^*(0)$. But $x = 1$ is not a solution of (P). This shows the failure when h is not closed.

Proposition 16. (e.g. [35, Theorem 23.5]) Let $g : \mathbb{R}^n \rightarrow \mathbb{R}$ be a proper closed convex function. Then $\bar{x} \in \partial g^*(\bar{y})$ if and only if \bar{x} is an optimal solution of (10).

The next result provides a consequence of assumption (iii).

Proposition 17. Let $h : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ be a polyhedral convex function such that $\dim \text{epi } h = n + 1$. Then $\text{epi } h^*$ has a vertex.

Proof. The assumption implies that there is a point $x \in \text{int dom } h$, such that h is affine in a neighborhood of x . Whence $\partial h(x)$ is a singleton set $\{\bar{y}\} = \{\nabla h(x)\}$. The Fenchel-Young inequality implies

$$\forall z \in \mathbb{R}^n : \quad h^*(z) \geq z^T x - h(x).$$

Using [35, Theorem 23.5], we see that equality holds if and only if $z = \bar{y}$. This means that

$$H = \{(z, r^*) \in \mathbb{R}^{n+1} \mid r^* = z^T x - h(x)\}$$

is a supporting hyperplane of $\text{epi } h^*$ such that $\text{epi } h^* \cap H = \{(\bar{y}, h^*(\bar{y}))\}$, i.e., $(\bar{y}, h^*(\bar{y}))$ is a vertex of $\text{epi } h^*$. \square

We obtain the following algorithm:

Algorithm 18. (1) Compute the vertices $(y^1, s^1), \dots, (y^k, s^k)$ of $\text{epi } h^*$ by solving a polyhedral projection problem, see Section 6.

(2) Determine (y^*, s^*) such that $s^* - g^*(y^*) = \min_{i=1, \dots, k} [s^i - g^*(y^i)]$,

(3) Compute an optimal solution \bar{x} of (10) with parameter y^* .

Result:

\bar{x} is an optimal solution of (P).

Theorem 19. Let the assumptions (i') to (vi') be satisfied. Then, Algorithm 18 works correctly.

Proof. Note first that (P) has a solution \bar{x} by Assumption (i'). A solution \bar{x} of (P) is by definition a point in $\text{dom } g \cap \text{dom } h$. Since h is polyhedral convex and $\bar{x} \in \text{dom } h$, we have $\partial h(\bar{x}) \neq \emptyset$, see e.g. [35, Theorem 23.10]. By Proposition 14 (i), a point $\bar{y} \in \partial h(\bar{x})$ is a solution to (D). Whence a solution of (D) exists. In Steps (1) and (2) of the algorithm, a solution \bar{y} of (D) is computed. To see this, we need to check the assumptions (i) to (iv) of Algorithm 12 in Section 4, reformulated for problem (D):

- (D) has an optimal solution, as shown above.
- h^* is polyhedral convex as so is h .
- $\text{epi } h^*$ has a vertex. This follows from Proposition 17.
- A representation A of h^* is known, as it can be obtained from a representation of h , see Example 5 in Section 2.

Since \bar{y} computed in Steps (1) and (2) is a solution to (D), we have $\bar{y} \in \text{dom } g^*$. Hence Problem (10) is bounded (by the definition of conjugates). In this case, by Assumption (vi'), an optimal solution \bar{x} of (10) exists. By Proposition 16, we get $\bar{x} \in \partial g^*(\bar{y})$. Proposition 14 (ii) implies that \bar{x} is an optimal solution of (P). \square

6 Polyhedral projection via multiple objective linear programming

For matrices $B \in \mathbb{R}^{m \times n}$, $C \in \mathbb{R}^{m \times k}$ and a vector $c \in \mathbb{R}^m$, we consider the problem

$$\text{compute } Y = \left\{ x \in \mathbb{R}^n \mid \exists u \in \mathbb{R}^k : Bx + Cu \geq c \right\}. \quad (\text{PP})$$

Solving (PP) means essentially to find a V-representation of Y , see [28] for details. Using the terminology of Section 2, (PP) provides a P-representation of Y . The polyhedron Y can be seen as the projection of the polyhedron

$$X = \left\{ (x, u) \in \mathbb{R}^n \times \mathbb{R}^k \mid Bx + Cu \geq c \right\} \quad (13)$$

into \mathbb{R}^n .

In [28], the equivalence between the projection problem (PP) and multiple objective linear programming (MOLP) is shown. In particular, [28, Theorem 3] yields that a V-representation of Y can be obtained by solving the multiple objective linear program

$$\min_{x \in \mathbb{R}^n, u \in \mathbb{R}^k} \begin{pmatrix} x \\ -e^T x \end{pmatrix} \quad \text{subject to} \quad Bx + Cu \geq c, \quad (\text{MOLP})$$

where $e = (1, \dots, 1)^T$.

A MOLP-solver such as *Bensolve* [27, 29] computes not only a solution to (MOLP) in the sense of [21, 26] but also a V-representation of the so-called *upper image*. The upper image of (MOLP) is the set

$$\mathcal{P} := \left\{ z \in \mathbb{R}^{n+1} \mid \exists u \in \mathbb{R}^k, \exists x \in \mathbb{R}^n : z \geq \begin{pmatrix} x \\ -e^T x \end{pmatrix}, Bx + Cu \geq c \right\},$$

whereas the *image* of (MOLP), in the literature also called *feasible set in the objective space*, is

$$Q := \left\{ z \in \mathbb{R}^{n+1} \mid \exists u \in \mathbb{R}^k, \exists x \in \mathbb{R}^n : z = \begin{pmatrix} x \\ -e^T x \end{pmatrix}, Bx + Cu \geq c \right\}.$$

Clearly, we have

$$\mathcal{P} = Q + \mathbb{R}_+^{n+1}$$

and

$$Q \subseteq \{z \in \mathbb{R}^{n+1} \mid e^T z = 0\}.$$

Thus, a V-representation of Q can be obtained from a V-representation of \mathcal{P} by omitting all points and directions x with $e^T z \neq 0$. Evidently, a V-representation of Y can be obtained from a V-representation of Q by omitting the last component of each vector.

7 Application and numerical results

First, we solve an instance of a location problem, then we compare our results with two algorithms from the literature and finally we present two new examples. We used the VLP-Solver Bensolve version 2.0.1 [27, 29] to compute the polyhedral projection problems. VLP stands for *vector linear programming* which is a generalization of MOLP. The remaining part was implemented in Matlab® R2015b. In particular, we use the `linprog` command if problem (10) needs to be solved, which reduces to an LP here. All computations were run on a computer with Intel® Core™ i5 CPU with 3.3 GHz.

As both g and h are polyhedral convex functions in our first two examples, we obtain two alternatives to solve the problems. Algorithm 12 developed in Section 4 (case g polyhedral) is referred to as the *primal algorithm*, whereas Algorithm 18 of Section 5 (case h polyhedral) is called the *dual algorithm*. As we will see, both algorithms are justified.

Example 20. The classical location theory is dedicated to the problem of locating a new facility such that all relevant distances, e.g. to customers, are minimized [7, 8, 11, 13, 19, 34, 41]. Nevertheless, since some facilities also cause negative effects like noise, stench or even danger, such facilities need to be established as far away as possible from nature reserves and residential areas. As examples one may think of an airport, railway station, industrial plant, dump site, power plant or wireless station.

The problem of locating such a so-called semi-obnoxious facility consists of minimizing the weighted sum of distances to the so-called attraction points (due to economical reasons) and to maximize the weighted sum of distances to the so-called repulsive ones (in order to avoid negative effects).

First attempts to solve location problems with such undesirable facilities appeared in the 1970's [4, 6, 17]. Algorithms based on a DC formulation are presented for instance in [3, 9, 30, 39].

It is reasonable to establish the new facility in a given area, city or state, which we assume to be a bounded polyhedral set $\mathcal{P} = \{x \in \mathbb{R}^n \mid Px \geq p\} \subseteq \text{dom } h$. The DC location problem can be formulated as

$$\min_{x \in \mathbb{R}^n} [g(x) - h(x)], \tag{L}$$

with functions $g, h : \mathbb{R}^n \rightarrow \mathbb{R}_+$ defined as

$$g(x) := \mathbb{I}_{\mathcal{P}}(x) + \sum_{m=1}^{\overline{M}} \overline{w}_m \gamma_{\overline{B}_m}(x - \overline{a}^m), \quad h(x) := \sum_{m=1}^{\underline{M}} \underline{w}_m \gamma_{\underline{B}_m}(x - \underline{a}^m), \quad (14)$$

where $\mathbb{I}_{\mathcal{P}}$ denotes the indicator function that attains the value 0, if $x \in \mathcal{P}$, and $+\infty$ otherwise. The parameters $\overline{a}^1, \dots, \overline{a}^{\overline{M}} \in \mathbb{R}^n$, ($\overline{M} \geq 1$) denote the attracting points with weights $\overline{w}_1, \dots, \overline{w}_{\overline{M}} > 0$, and $\underline{a}^1, \dots, \underline{a}^{\underline{M}} \in \mathbb{R}^n$, ($\underline{M} \geq 1$) denote the repulsive points with weights $\underline{w}_1, \dots, \underline{w}_{\underline{M}} > 0$. The polyhedral unit balls $\overline{B}_1, \dots, \overline{B}_{\overline{M}}$ and $\underline{B}_1, \dots, \underline{B}_{\underline{M}}$ induce gauge distances that we associate with $\overline{a}^1, \dots, \overline{a}^{\overline{M}}$ and $\underline{a}^1, \dots, \underline{a}^{\underline{M}}$, respectively.

For a compact polyhedral set $B \subseteq \mathbb{R}^n$, which contains the origin in its interior, the polyhedral gauge distance $\gamma_B : \mathbb{R}^n \rightarrow \mathbb{R}$ from $a \in \mathbb{R}^n$ to $x \in \mathbb{R}^n$ is defined as $\gamma_B(x - a) := \min \{ \lambda \geq 0 \mid x - a \in \lambda B \}$, see [11, 35].

The epigraph of g is given by

$$\begin{aligned} \text{epi } g &= \left\{ (x, r) \in \mathcal{P} \times \mathbb{R} \mid r \geq \sum_{i=1}^{\overline{M}} \overline{w}_i \gamma_{\overline{B}_i}(x - \overline{a}^i) \right\} \\ &= \left\{ (x, r) \in \mathcal{P} \times \mathbb{R} \mid r \geq \sum_{i=1}^{\overline{M}} \overline{w}_i \min \{ \lambda_i \geq 0 \mid x - \overline{a}^i \in \lambda_i \overline{B}_i \} \right\} \\ &= \left\{ (x, r) \in \mathcal{P} \times \mathbb{R} \mid \exists \lambda \in \mathbb{R}_+^{\overline{M}} : r \geq \sum_{i=1}^{\overline{M}} \lambda_i \overline{w}_i, x - \overline{a}^i \in \lambda_i \overline{B}_i \right\} \\ &= \left\{ (x, r) \in \mathbb{R}^{n+1} \mid \exists \lambda \in \mathbb{R}^{\overline{M}} : r \geq \sum_{i=1}^{\overline{M}} \lambda_i \overline{w}_i, \hat{\overline{B}}_i(x - \overline{a}^i) \leq \lambda_i e; Px \geq p; \lambda \geq 0 \right\}, \end{aligned}$$

where $\hat{\overline{B}}_1, \dots, \hat{\overline{B}}_{\overline{M}}$ are the matrices defining the polyhedral unit balls $\overline{B}_1, \dots, \overline{B}_{\overline{M}}$, respectively, i.e., $x \in \overline{B}_i$ if and only if $\hat{\overline{B}}_i x \leq e$ for $(i = 1, \dots, \overline{M})$.

A representation $\overline{A} = (\overline{B}, \overline{b}, \overline{C}, \overline{c})$ of g is given by

$$\overline{B} = \begin{pmatrix} -\hat{\overline{B}}_1 \\ \vdots \\ -\hat{\overline{B}}_{\overline{M}} \\ 0 \\ \vdots \\ 0 \\ 0 \\ P \end{pmatrix}, \quad \overline{b} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \end{pmatrix}, \quad \overline{C} = \begin{pmatrix} e & & & \\ & \ddots & & \\ & & e & \\ \hline 1 & & & \\ & \ddots & & \\ & & 1 & \\ \hline -\overline{w}_1 & \dots & -\overline{w}_{\overline{M}} \\ 0 & \dots & 0 \end{pmatrix}, \quad \overline{c} = \begin{pmatrix} -\hat{\overline{B}}_1 \overline{a}^1 \\ \vdots \\ -\hat{\overline{B}}_{\overline{M}} \overline{a}^{\overline{M}} \\ 0 \\ \vdots \\ 0 \\ 0 \\ p \end{pmatrix}.$$

A representation of h is given likewise.

The epigraphs $\text{epi } g^*$ and $\text{epi } h^*$ and corresponding representations can easily be obtained by applying Example 5 and Proposition 6. Moreover, the assumptions (i)-(iv) in Section 4

(case g polyhedral) and (i')-(vi') in Section 5 (case h polyhedral) are satisfied and hence, Algorithms 12 and 18 can be applied in order to solve the location problem (L) .

We note further that the conjugates of g and h can be written as

$$g^*(y) = \min \left\{ \sum_{i=1}^{\overline{M}} \left[\underline{a}^{iT} y^i + \mathbb{I}_{\underline{w}_i \underline{B}_i^*}(y^i) \right] + \sigma_{\mathcal{P}}(y^0) \mid \sum_{i=0}^{\overline{M}} y^i = y \right\} \quad (15)$$

$$h^*(y) = \min \left\{ \sum_{i=1}^{\underline{M}} \left[\underline{a}^{iT} y^i + \mathbb{I}_{\underline{w}_i \underline{B}_i^*}(y^i) \right] \mid \sum_{i=1}^{\underline{M}} y^i = y \right\} \quad (16)$$

[40], where $B^* := \{y \in \mathbb{R}^n \mid \forall x \in B : x^T y \leq 1\}$ defines the dual unit ball of B . If B is polyhedral then so is B^* .

We next solve several instances of (L) in the plane. In dependence of the number of attracting and repulsive points, Tables 1 and 2 show the computational results of the primal and dual algorithm, respectively. In most cases, the primal algorithm performs better.

Remark 21. In Step (2) of Algorithm 18, $g^*(y^i)$ is calculated for all vertices (y^i, s^i) , $i = 1, \dots, k$, of $\text{epi } h^*$, obtained in Step (1). In general, we cannot directly calculate these objective values, see for example the representation (15), which constitutes a linear program. In the general setting, the convex program (10) needs to be solved for each vertex. On the other hand, in Step (2) of Algorithm 12, we need to calculate $h(x^i)$ for all vertices (x^i, r^i) , $i = 1, \dots, k$, of $\text{epi } g$, obtained in Step (1). If (in case both g and h are polyhedral) only a representation of h (i.e. a P -representation $\text{epi } h$) is given, a general procedure to get the values $h(x^i)$ is to solve the linear program $\min r$ s.t. $(x^i, r) \in \text{epi } h$. However, in the present example (and often in practice) one can directly calculate these objective values, i.e. there is no additional linear program to be solved. This is an advantage of the primal algorithm for our example from locational analysis.

Nevertheless, we also observe for the primal algorithm that an increasing number of attracting points influences the running time much more than an increasing number of repulsion points. It is vices versa for the dual algorithm, which leads to settings where the dual algorithm performs even better, such as $\overline{M} = 100$, $\underline{M} = 20$. Thus, both algorithms have their justification.

\overline{M}	\underline{M}						
	20	40	60	80	100	1000	5000
20	0.37	0.51	0.53	0.47	0.86	5.16	52.55
40	3.36	2.89	5.77	4.88	6.77	22.07	176.67
60	17.92	23.29	34.54	25.34	37.55	91.24	462.10
80	74.16	96.38	117.54	154.28	152.81	291.03	899.93
100	292.85	337.41	289.85	342.32	387.56	945.23	1797.00

Table 1: Computational results (running time in seconds) for Example 20 using the primal algorithm (case g polyhedral).

\overline{M}	\underline{M}				
	20	40	60	80	100
20	3.84	16.64	63.50	198.76	573.40
40	10.13	40.87	111.28	358.97	915.98
60	18.17	103.39	216.55	593.61	1436.48
80	44.35	192.98	465.88	943.77	1778.65
100	58.85	338.85	688.54	1316.08	2793.30
1000	91.52	406.74	1005.94	2065.33	4503.63
5000	2077.18	8082.09	-	-	-

Table 2: Computational results (running time in seconds) for Example 20 using the dual algorithm (case h polyhedral).

Example 22. Ferrer, Bagirov and Baliakov [15] introduce the DC extended cutting angle method (DCECAM) and compare it with the DC prismatic algorithm (DCPA) described in [14]. We compare Algorithms 12 and 18 with these two methods by means of Example 10 in [15]: Consider a DC programming problem (P) with

$$g(x) = |x_1 - 1| + 200 \sum_{i=2}^n \max \{0, |x_{i-1}| - x_i\}$$

and

$$h(x) = 100 \sum_{i=2}^n (|x_{i-1}| - x_i).$$

In Table 3 we present numerical results for the primal and dual algorithm developed in this article and compare them with the results of DCECAM and DCPA obtained in [15]. We see that both of our algorithms perform better than DCECAM and DCPA, where it should be noted, that the latter algorithms are designed for a more general problem class (non-polyhedral DC problems). In particular, the dual method is preferable for this example. Note that for $n = 10$ we need to solve a multiple linear program with $n + 2 = 12$ objectives.

Example 23. Let $Q \in \mathbb{R}^{n \times n}$ be a positive semi-definite symmetric matrix. The problem to maximize the convex quadratic function $h : \mathbb{R}^n \rightarrow \mathbb{R}$, $h(x) = x^T Q x$ over a polyhedral convex set $S \subseteq \mathbb{R}^n$ can be reformulated as a DC-program by letting $g = \mathbb{I}_S$, i.e. the indicator function of S , and minimizing $f = g - h$. For example, let $S := \{x \in \mathbb{R}^n \mid -e \leq x \leq e\}$ be an n -dimensional hypercube. As h is convex, Q has only non-negative eigenvalues. We assume that Q has at most $m \leq n$ positive eigenvalues. Equivalently, Q can be expressed as $Q = P^T P$ for some $P \in \mathbb{R}^{m \times n}$. For example, let the entries of P be defined by

$$P_{ij} = \text{floor}(m \cdot \sin((j - 1) \cdot m + i)),$$

where $\text{floor}(x) := \max \{z \in \mathbb{Z} \mid z \leq x\}$. Setting $y := Px$ and $T = \{y \in \mathbb{R}^m \mid \exists x \in S : y = Px\}$, we obtain the equivalent problem to minimize $\bar{h} : \mathbb{R}^m \rightarrow \mathbb{R} \cup \{+\infty\}$, $\bar{h}(y) = y^T y$ over the

n	DCECAM [15]	DCPA [15]	primal method	dual method
2	0.21	0.22	0.04	0.03
3	3.57	4.63	0.09	0.04
4	2.47	0.78	0.29	0.05
5	345.12	502.29	2.05	0.09
6	-	-	65.76	0.32
7	-	-	5107.84	1.17
8	-	-	-	47.36
9	-	-	-	247.21
10	-	-	-	3028.10

Table 3: Computational results for Example 10 in [15]. The running times (in seconds) for DCECAM and DCPA are taken from [15] and are obtained on a computer with Intel® Core™ i5-3470S CPU with 2.9 GHz.

polyhedron $T \subseteq \mathbb{R}^m$. For $n = 10$ and $m = 4$ we have

$$P = \begin{pmatrix} 3 & -4 & 1 & 1 & -4 & 3 & -1 & -3 & 3 & -3 \\ 3 & -2 & -3 & 3 & -4 & -1 & 3 & -4 & 2 & 1 \\ 0 & 2 & -4 & 2 & 0 & -4 & 3 & -2 & -2 & 3 \\ -4 & 3 & -3 & -2 & 3 & -4 & 1 & 2 & -4 & 2 \end{pmatrix}$$

and the optimal solution obtained by Algorithm 12 applied to the reformulated problem is

$$x = (1, -1, 1, 1, -1, 1, -1, -1, 1, -1)^T.$$

In Table 4, further numerical results for this example can be found.

n	m				
	2	3	4	5	6
10	0.02	0.04	0.6	8.7	76.0
50	0.02	1.57	22.5	-	-
200	0.04	2.39	62440.0	-	-
1000	0.15	3.10	-	-	-
5000	1.50	13.60	-	-	-

Table 4: Running times in seconds for Example 23.

Example 24. Let $g : \mathbb{R}^n \rightarrow \mathbb{R}$, $g(x) = x^T Q x$ for some positive definite symmetric matrix Q . Then the conjugate is $g^*(y) = \frac{1}{4} y^T Q^{-1} y$ and the optimal solution of (10) is $x = \frac{1}{2} Q^{-1} y$. For example, let $Q = P^T P$ where $P \in \mathbb{R}^{n \times n}$ defined by $P_{ij} = 1$ for $i \geq j$ and $P_{ij} = 0$ otherwise, then Q is positive definite. Further, we set

$$h(x) = \sum_{i=2}^n (|x_{i-1}| - x_i).$$

Some numerical results of minimizing $f = g - h$ using Algorithm 18 are shown in Table 5.

n	2	3	4	5	6	7	8	9	10
optimal value	-1.25	-2.75	-3.75	-4.75	-5.75	-6.75	-7.75	-8.75	-9.75
running time (sec)	0.02	0.02	0.02	0.02	0.05	0.21	1.85	21.5	281

Table 5: Numerical results for Example 24.

8 Conclusion

It is demonstrated that MOLP solvers provide a very easy way to solve DC programs where at least one of the two functions g, h in the objective $f = g - h$ is polyhedral. In case of both g and h being polyhedral, the investigations result in two different algorithms, called primal and dual algorithm. The running time of the primal algorithm depends mainly on the effort to compute the vertices of $\text{epi } g$ by the MOLP solver. Likewise the dual algorithm depends on $\text{epi } h^*$. The proposed methods work well, if the dimension of $\text{epi } g$ in Algorithm 12 or $\text{epi } h^*$ in Algorithm 18 is not too high (up to 11 in the examples above).

References

- [1] A. D. Aleksandrov. On the surfaces representable as difference of convex functions. (*Russian*) *Izvestiya Akad. Nauk Kazah. SSR*, 3:3–20, 1949.
- [2] H. Benson. An outer approximation algorithm for generating all efficient extreme points in the outcome set of a multiple objective linear programming problem. *Journal of Global Optimization*, 13:1–24, 1998.
- [3] P. C. Chen, P. Hansen, B. Jaumard, and H. Tuy. Weber’s problem with attraction and repulsion. *Journal of Regional Science*, 32:467–486, 1992.
- [4] R. L. Church and R. S. Garfinkel. Locating an obnoxious facility on a network. *Transportation Science*, 12(2):107–118, 1978.
- [5] L. Csirmaz. Using multiobjective optimization to map the entropy region. *Computational Optimization and Applications*, 63(1):45–67, 2016.
- [6] B. Dasarathy and L. J. White. A maxmin location problem. *Operations Research*, 28(6):1385–1401, 1980.
- [7] Z. Drezner, editor. *Facility location: A survey of applications and methods*. Springer Series in Operations Research. Springer-Verlag, New York, 1995.
- [8] Z. Drezner, K. Klamroth, A. Schöbel, and G. O. Wesolowsky. The weber problem. In Z. Drezner and H. Hamacher, editors, *Facility Location - Applications and Theory*. Springer-Verlag, 2002.

- [9] Z. Drezner and G. O. Wesolowsky. The Weber problem on the plane with some negative weights. *INFOR*, 29(2):87–99, 1991.
- [10] M. Dür. Conditions characterizing minima of the difference of functions. *Monatshefte für Mathematik*, 134(4):295–303, 2002.
- [11] R. Durier and C. Michelot. Geometrical properties of the Fermat-Weber problem. *European Journal of Operational Research*, 20(3):332–343, 1985.
- [12] M. Ehrgott, A. Löhne, and L. Shao. A dual variant of Benson’s outer approximation algorithm. *Journal of Global Optimization*, 52(4):757–778, 2012.
- [13] H. A. Eiselt and G. Laporte. Objectives in location problems. In Z. Drezner, editor. *Facility Location, a Survey of Applications and Methods*, Springer Series in Operations Research. Springer, 1995.
- [14] A. Ferrer. Applying global optimization to a problem in short-term hydrothermal scheduling. In *Generalized convexity, generalized monotonicity and applications. Proceedings of the 7th international symposium on generalized convexity and generalized monotonicity, Hanoi, Vietnam, August 27–31, 2002.*, pages 263–285. New York, NY: Springer, 2005.
- [15] A. Ferrer, A. Bagirov, and G. Beliakov. Solving dc programs using the cutting angle method. *Journal of Global Optimization*, 61(1):71–89, 2014.
- [16] F. Flores-Bazán. On minima of the difference of functions. *Journal of Optimization Theory and Applications*, 93(3):525–531, 1997.
- [17] A. J. Goldman and P. M. Dearing. Concepts of optimal location for partially noxious facilities. *Bulletin of the Operations Research Society of America*, 23(1):B–31, 1975.
- [18] O. Güler. *Foundations of optimization*. New York, NY: Springer, 2010.
- [19] H. W. Hamacher. *Mathematische Lösungsverfahren für planare Standortprobleme*. Vieweg, 1995.
- [20] P. Hartman. On functions representable as a difference of convex functions. *Pacific Journal of Mathematics*, 9(3):707–713, 1959.
- [21] F. Heyde and A. Löhne. Solution concepts in vector optimization: a fresh look at an old story. *Optimization*, 60(12):1421–1440, 2011.
- [22] J.-B. Hiriart-Urruty. A general formula on the conjugate of the difference of functions. *Canadian Mathematical Bulletin*, 29(4):482–485, 1986.
- [23] R. Horst and N. V. Thoai. DC programming: overview. *Journal of Optimization Theory and Applications*, 103(1):1–43, 1999.
- [24] C. Lassez and J. Lassez. *Quantifier Elimination for Conjunctions of Linear Constraints Via a Convex Hull Algorithm*. IBM Thomas J. Watson Research Division, 1990.
- [25] B. Lemaire and M. Volle. Duality in DC programming. In *Generalized convexity, generalized monotonicity: recent results (Luminy, 1996)*, volume 27 of *Nonconvex Optim. Appl.*, pages 331–345. Kluwer Academic Publishers, Dordrecht, 1998.

- [26] A. Löhne. *Vector Optimization with Infimum and Supremum*. Vector Optimization. Springer, Berlin, 2011.
- [27] A. Löhne and B. Weißing. Bensolve - VLP solver, version 2.0.1. www.bensolve.org.
- [28] A. Löhne and B. Weißing. Equivalence between polyhedral projection, multiple objective linear programming and vector linear programming. *Mathematical Methods of Operations Research*, 2016. DOI: 10.1007/s00186-016-0554-0.
- [29] A. Löhne and B. Weißing. The vector linear program solver Bensolve – notes on theoretical background. *European Journal of Operational Research*, 2016. DOI: 10.1016/j.ejor.2016.02.039.
- [30] C. D. Maranas and C. Floudas. A global optimization method for Weber’s problem with attraction and repulsion. In W. Hager, D. Hearn, and P. Pardalos, editors, *Large Scale Optimization*, pages 259–285. Springer US, 1994.
- [31] J. E. Martínez-Legaz and A. Seeger. A formula on the approximate subdifferential of the difference of convex functions. *Bulletin of the Australian Mathematical Society*, 45(1):37–41, 1992.
- [32] J. E. Martínez-Legaz and I. Singer. An extension of d. c. duality theory, with an appendix on \ast -subdifferentials. *Optimization*, 42(1):9–37, 1997.
- [33] T. Motzkin, H. Raiffa, G. Thompson, and R. Thrall. The double description method. Contrib. Theory of Games, II, Ann. Math. Stud. No. 28, 51-73 (1953)., 1953.
- [34] S. Nickel. *Location Theory: A Unified Approach*. Springer-Verlag, 2005.
- [35] R. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, 1972.
- [36] I. Singer. A Fenchel-Rockafellar type duality theorem for maximization. *Bulletin of the Australian Mathematical Society*, 20(2):193–198, 1979.
- [37] J. F. Toland. Duality in nonconvex optimization. *Journal of Mathematical Analysis and Applications*, 66(2):399–415, 1978.
- [38] H. Tuy. *Convex analysis and global optimization*, volume 22 of *Nonconvex Optimization and its Applications*. Kluwer Academic Publishers, Dordrecht, 1998.
- [39] H. Tuy, F. Al-Khayyal, and F. Zhou. A d.c. optimization method for single facility location problems. *Journal of Global Optimization*, 7(2):209–227, 1995.
- [40] A. Wagner, J. E. Martinez-Legaz, and C. Tammer. Locating a semi-obnoxious facility – a toland-singer duality based approach. *Journal of Convex Analysis*, 23(4), 2016.
- [41] A. Weber. *Über den Standort der Industrien. Erster Teil: Reine Theorie des Standortes*. Tübingen: Mohr, 1909.